Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Canceled).

2. (Previously presented) The method of claim 23, further comprising:

forming an ordered set of deadlock-free sub-topologies of said network, each sub-topology comprising links that are not used in any other sub-topology; and

generating said routing table in response to said ordered set of deadlock-free sub-topologies.

3. (Original) The method of claim 2, wherein said forming said ordered set of deadlock-free sub-topologies of said network further comprises forming at least one sub-topology of said network that is a spanning layer of said network.

4. (Original) The method of claim 3, further comprising maintaining a cost of a corresponding link between each of said nodes in said network within each entry of said routing table.

5. (Original) The method of claim 4, wherein said adding said column to said routing table further comprises:

for each entry within said column, performing the following steps

determining a set of cost values, wherein each value within said set of cost values reflects a sum of the cost of reaching a selected neighbor node of said node from said corresponding forwarding node and the cost of reaching said node from said selected neighbor node,

determining a minimum value of said set of cost values, and

determining forwarding information for said entry indicating said selected neighbor node associated with said minimum value.

6. (Original) The method of claim 4, wherein said adding said row to said routing table further comprises:

for each entry within said row, performing the following steps

determining a set of cost values, wherein each value within said set of cost values reflects a sum of the cost of reaching said corresponding destination node from a selected neighbor node of said node and the cost of reaching said selected neighbor node from said node,

determining a minimum value of said set of cost values, and

determining forwarding information for said entry indicating said selected neighbor node associated with said minimum value.

7. (Canceled).

8. (Previously presented) The system of claim 24, wherein said routing logic is further operable to:

form an ordered set of deadlock-free sub-topologies of said network, each sub-topology comprising links that are not used in any other sub-topology; and

generate said routing table in response to said ordered set of deadlock-free sub-topologies.

9. (Original) The system of claim 8, wherein said routing logic is further operable to form said ordered set of deadlock-free sub-topologies of said network further by forming at least one sub-topology of said network that is a spanning layer of said network.

10. (Original) The system of claim 9, wherein said routing logic is further operable to maintain a cost of a corresponding link between each of said nodes in said network within each entry of said routing table.

11. (Original) The system of claim 10, wherein routing logic operable to add said column to said routing table is further operable to perform the following steps for each entry within said column:

determine a set of cost values, wherein each value within said set of cost values reflects a sum of the cost of reaching a selected neighbor node of said node from said corresponding forwarding node and the cost of reaching said node from said selected neighbor node;

determine a minimum value of said set of cost values; and

determine forwarding information for said entry indicating said selected neighbor node associated with said minimum value.

12. (Original) The system of claim 10, wherein said routing logic operable to add said row to said routing table is further operable to perform the following steps for each entry within said row:

determine a set of cost values, wherein each value within said set of cost values reflects a sum of the cost of reaching a corresponding node from a selected neighbor node of said node and the cost of reaching said selected neighbor node from said node;

determine a minimum value of said set of cost values; and

determine forwarding information for said entry indicating said selected neighbor node associated with said minimum value.

13. (Previously presented) The system of claim 24, wherein said routing logic comprises at least one digital logic circuit.

14. (Previously presented) The system of claim 24, wherein said routing logic comprises program code loaded into a memory of a computer system.

Claims 15 – 17: Canceled

18. (Withdrawn) A method for inserting routing information regarding a node into a routing table, wherein said routing table defines a deadlock-free set of paths through a network of nodes connected by a plurality of links, comprising:

identifying an ordered set of layers within the network, each layer comprising a deadlock-free sub-topology within the network that is composed of links that are not part of any other layer;

obtaining identification of at least one link operable to deliver data to said node;

obtaining identification of at least one link operable to convey data transmitted from said node;

storing said identification of said at least one link operable to deliver data to said node into a new highest layer within ~~an~~ the ordered set of layers ~~upon~~ from which said deadlock-free set of paths are derived;

storing said identification of said at least one link operable to convey ~~day~~ data transmitted from said node into a new lowest layer within said ordered set of layers upon which said deadlock-free set of paths are derived; and

adding routing information to said forwarding table that reflects said new lowest layer and said new highest layer.

19. (Withdrawn) A method for inserting routing information regarding a unidirectional link into a routing table, wherein said routing table defines a deadlock-free set of paths through a network of nodes connected by links, comprising:

identifying an ordered set of layers within the network, each layer comprising a deadlock-free sub-topology within the network that is composed of links that are not part of any other layer;

adding said unidirectional link into a new layer within ~~an~~ the ordered set of layers upon from which said deadlock-free set of paths are derived; and

recalculating said deadlock-free set of paths in response to said adding of said unidirectional link into said new layer.

20. (Withdrawn) A method for inserting routing information regarding a bi-directional link into a routing table, wherein said routing table defines a deadlock-free set of paths through a network of nodes connected by links, wherein said bi-directional link is between a first node and a second node, comprising:

identifying an ordered set of layers within the network, each layer comprising a deadlock-free sub-topology within the network that is composed of links that are not part of any other layer;

determining a first unidirectional link from said first node to said second node;

determining a second unidirectional link from said second node to said first node;

adding said first unidirectional link to a lowest layer within ~~an~~ the ordered set of layers upon from which said deadlock-free set of paths are derived;

adding said second unidirectional link to a highest layer within said ordered set of layers upon from which said deadlock-free set of paths are derived; and

recalculating said deadlock-free set of paths.

21. (Previously presented) The method of claim 23, further comprising iteratively performing said steps of adding a row of entries and adding a column of entries in order to add routing information to said routing table for a plurality of nodes.

22. (Previously presented) The method of claim 23, wherein said existing deadlock-free set of paths are through a network of two nodes.

23. (Previously presented) A method for adding routing information for a new node to a routing table with a plurality of entries that reflect an existing deadlock-free set of paths through a network of nodes, wherein the routing table has a row for each source node in the network and a column for each destination node in the network and wherein a table entry located at an entry row and an entry column

identifies a link that can be used to send data from the source node in the entry row to the destination node in the entry column, the method comprising:

adding to the routing table, a row including a plurality of entries, each entry identifying a link that directly connects the new node to a neighbor node that can be connected, via existing deadlock-free paths described by the table, to a destination node associated with the entry column; and

adding to the routing table a column including a plurality of entries, each entry identifying a link that can be used to connect a source node associated with the entry row, via existing deadlock-free paths described by the table, to a neighbor node that can be directly connected to the new node.

24. (Previously presented) A system for adding routing information for a new node to a routing table with a plurality of entries that reflect an existing deadlock-free set of paths through a network of nodes, wherein the routing table has a row for each source node in the network and a column for each destination node in the network and wherein a table entry located at an entry row and an entry column identifies a link that can be used to send data from the source node in the entry row to the destination node in the entry column, comprising routing logic operable to:

add to the routing table, a row including a plurality of entries, each entry identifying a link that directly connects the new node to a neighbor node that can be connected, via existing deadlock-free paths described by the table, to a destination node associated with the entry column; and

add to the routing table a column including a plurality of entries, each entry identifying a link that can be used to connect a source node associated with the entry row, via existing deadlock-free paths described by the table, to a neighbor node that can be directly connected to the new node.

25. (Previously presented) A system for adding routing information for a new node to a routing table with a plurality of entries that reflect an existing deadlock-free set of paths through a network of nodes, wherein the routing table has a row for each

source node in the network and a column for each destination node in the network and wherein a table entry located at an entry row and an entry column identifies a link that can be used to send data from the source node in the entry row to the destination node in the entry column, comprising:

means for adding to the routing table, a row including a plurality of entries, each entry identifying a link that directly connects the new node to a neighbor node that can be connected, via existing deadlock-free paths described by the table, to a destination node associated with the entry column; and

means for adding to the routing table a column including a plurality of entries, each entry identifying a link that can be used to connect a source node associated with the entry row, via existing deadlock-free paths described by the table, to a neighbor node that can be directly connected to the new node.

26. (Previously presented) A computer program product including a computer readable medium, said computer readable medium having a computer program stored thereon, said computer program for adding routing information for a node to a routing table, wherein said routing table includes routing information reflecting an existing deadlock-free set of paths through a network of nodes, said computer program comprising:

program code for adding to the routing table, a row including a plurality of entries, each entry identifying a link that directly connects the new node to a neighbor node that can be connected, via existing deadlock-free paths described by the table, to a destination node associated with the entry column; and

program code for adding to the routing table a column including a plurality of entries, each entry identifying a link that can be used to connect a source node associated with the entry row, via existing deadlock-free paths described by the table, to a neighbor node that can be directly connected to the new node.

Claim 27: Canceled